

(RESEARCH ARTICLE)



STABILITY ANALYSIS AND PID CONTROL OPTIMIZATION OF A HYBRID VTOL UAV DURING HOVER-TO-FORWARD FLIGHT TRANSITION USING MATLAB/SIMULINK

Solomon Saiki *, Ikeru Oluchi Olukayode, Ekezoka Dada Evieshoyan, Samuel Agebe Ajeh, Danjuma Amos Olobo and Osheku Gideon Afoyani.

National space research and development agency, BATCSTP Lagos Nigeria.

* Corresponding Author

Global Journal of Engineering and Technology Advances, 2026, 29(01), 306–318

Publication history: Received on 14 September 2026; revised on 20 October 2026; accepted on 22 October 2026

Article DOI: <https://doi.org/10.30574/gjeta.2026.29.1.0305>

Copyright © 2026 Author(s) retain the copyright of this article. This article is published under the terms of the Creative Commons Attribution License 4.0

ABSTRACT

Hybrid vertical take-off and landing (VTOL) unmanned aerial vehicles combine the hovering capability of multi-rotors with the endurance of fixed-wing aircraft, but the hover-to-forward-flight transition is the most demanding phase of the mission: the dominant source of lift migrates from the lift rotors to the wing while the pitch dynamics change from marginally unstable to aerodynamically stabilized. This paper presents a stability analysis and a PID control optimization for a quad-plane hybrid VTOL UAV during this transition. A nonlinear longitudinal three-degree-of-freedom model with rotor drag, rotor flapping moment, stall-blended aerodynamics and actuator dynamics is developed in a form that maps directly onto a MATLAB/Simulink block diagram. The model is linearized at six trim airspeeds between 0 and 22 m/s, showing an unstable oscillatory pair at hover ($\sigma = +0.80 \text{ s}^{-1}$, $\omega = 1.41 \text{ rad/s}$) and an unstable real pole of $+0.99 \text{ s}^{-1}$ at 5 m/s, which disappear as dynamic pressure grows. A gain-scheduled cascade of three PID loops (attitude, altitude and airspeed) with airspeed-based control allocation between lift rotors and elevator is then tuned by particle swarm optimization (PSO) using a cost that combines altitude and attitude ITAE, airspeed tracking, control effort and a phase-margin constraint of 45° . Compared with a manually tuned baseline, the optimized controller reduces the peak altitude deviation during transition from 8.0 m to 3.7 m, the altitude ITAE by 6.3 times, and the airspeed overshoot from 5.8% to 2.6%, while keeping the pitch-loop phase margin at or above 45° at every airspeed. In a 40-run Monte Carlo study with $\pm 15\%$ mass, reduced static margin, inertia and drag uncertainty and a 3 m/s vertical gust, all runs were stable for both controllers, and the optimized controller kept the worst-case altitude deviation to 8.6 m against 20.7 m for the baseline.

Keywords: Hybrid VTOL UAV; Transition Flight; Stability Analysis; PID Control; Particle Swarm Optimization; MATLAB/Simulink; Gain Scheduling

1 INTRODUCTION

Unmanned aerial vehicles that can take off and land vertically without a runway while cruising efficiently on a wing are attractive for logistics, mapping, surveillance and search-and-rescue. Surveys of the field [1], [2] show that the quad-plane (or “lift + cruise”) configuration, a fixed-wing airframe with a set of dedicated vertical lift rotors and a separate pusher propeller, is among the most widely adopted because it avoids the mechanical complexity of tilting mechanisms.

Its price is a flight-control problem with two distinct regimes: a multi rotor-like regime in which attitude is controlled by differential rotor thrust, and a fixed-wing regime in which aerodynamic surfaces generate the control moments.

The transition between these regimes is where most difficulties arise. During the maneuver the airspeed rises from zero to cruise, the wing lift grows with the square of airspeed, the rotor thrust must be withdrawn at a matching rate to hold altitude, and the angle of attack must be kept below stall. The pitch dynamics change qualitatively: at hover the vehicle is unstable in the sense of an inverted-pendulum-like pitch/translation coupling, whereas in cruise the aerodynamic stiffness and damping produce a well-damped short-period mode [6], [7]. Control-surface effectiveness is negligible at low speed and rotor authority is no longer available when the rotors are shut down, so control allocation must be scheduled. Studies of transition modeling and control [3] and recent reviews [2] emphasize that this scheduling, rather than the individual loops, governs transition quality.

PID control remains the industry-standard approach for small UAVs because it is simple, transparent and certifiable [5], [8], [9]. Its weakness is tuning: classical rules such as Ziegler–Nichols [10] are based on a single operating point and offer no guarantee of performance across a transition in which the plant changes by an order of magnitude in bandwidth. Metaheuristic optimizers such as particle swarm optimization (PSO) [11] and differential evolution [12] provide a convenient way to tune several loops simultaneously against a time-domain cost computed in simulation, but a purely time-domain cost can produce gains that perform well in simulation and have very little stability margin, a problem demonstrated in Section 5.3.

The contributions of this paper are: (i) a nonlinear longitudinal model of a hybrid VTOL UAV suitable for implementation in MATLAB/Simulink, including the rotor effects that destabilize the low-speed regime; (ii) a stability analysis based on linearization at trim points spanning the transition, including eigenvalue and phase-margin maps; (iii) a PSO-based optimization of nine PID gains that includes a phase-margin constraint in the cost; and (iv) a performance and robustness comparison against a manually tuned baseline under nominal, gust and parameter-uncertainty conditions.

2 VEHICLE MODEL

2.1 Configuration and assumptions

The vehicle is a 12 kg quad-plane with a 0.9 m^2 wing, four vertical lift rotors with a combined maximum thrust of 180 N (thrust-to-weight ratio 1.5), a rear pusher propeller with a maximum thrust of 40 N, and a conventional elevator. The rotors are fixed relative to the airframe. Because the focus is the pitch/altitude/airspeed behavior during transition, the motion is restricted to the longitudinal plane (three degrees of freedom plus position), lateral-directional dynamics are decoupled, and the atmosphere is assumed to be standard sea-level ($\rho = 1.225 \text{ kg/m}^3$). Parameters are listed in Table 2.

Table 1: Vehicle and model parameters

Parameter	Value
Mass m	12.0 kg
Pitch inertia I_{yy}	$1.2 \text{ kg}\cdot\text{m}^2$
Wing area S / chord c	0.90 m^2 / 0.30 m
C_{L0} / $C_{L\alpha}$	0.25 / 5.0 rad^{-1}
C_{D0} / induced factor k	0.060 / 0.060
C_{m0} / $C_{m\alpha}$ / C_{mq} / $C_{m\delta e}$	0.02 / -1.20 / -12.0 / -1.10
Stall angle α_s (blend M)	0.28 rad (16.0°), $M = 50$
Max lift-rotor thrust / pusher thrust	180 N / 40 N
Max elevator / rotor moment	$\pm 25^\circ$ / $\pm 12 \text{ N}\cdot\text{m}$
Actuator lags (rotor, pusher, elevator, rotor moment)	0.08, 0.15, 0.05, 0.08 s
Rotor drag k_{rd} / flapping moment k_{fl}	$0.8 \text{ N}\cdot\text{s/m}$ / $0.8 \text{ N}\cdot\text{m}\cdot\text{s/m}$ (at full rotor thrust)
Rotor-to-wing lift hand-over	5 m/s to 18 m/s
Rotor-to-elevator pitch hand-over	6 m/s to 14 m/s

2.2 Equations of motion

With body-axis velocity components u (forward) and w (downward), pitch rate q and pitch angle θ , the longitudinal equations of motion are [4], [6]:

$$\dot{u} = -q w - g \sin \theta + F_x / m \quad (1)$$

$$\dot{w} = q u + g \cos \theta + F_z / m \quad (2)$$

$$\dot{q} = M / I_{yy}, \quad \dot{\theta} = q \quad (3)$$

$$\dot{h} = u \sin \theta - w \cos \theta \quad (4)$$

where the body-axis forces and the pitching moment are assembled from aerodynamic, rotor and pusher contributions:

$$F_x = -D \cos \alpha + L \sin \alpha + T_f - k_{rd} (T_v / T_{v,max}) u \quad (5)$$

$$F_z = -D \sin \alpha - L \cos \alpha - T_v - k_{rd} (T_v / T_{v,max}) w \quad (6)$$

$$M = \bar{q} S c C_m + M_r + k_{fl} (T_v / T_{v,max}) u \quad (7)$$

Here $\bar{q} = \frac{1}{2} \rho V^2$ is the dynamic pressure, $\alpha = \text{atan2}(w, u)$, and L and D are the lift and drag. The two terms proportional to $T_v / T_{v,max}$ represent, respectively, rotor momentum drag and the rotor flapping moment. The latter is a nose-up moment that grows with forward speed, which is the well-known speed instability of rotors in translational flight and is the reason the low-speed regime is unstable in this model.

2.3 Aerodynamics and actuators

Lift is computed from $C_L(\alpha)$, a linear curve $C_{L0} + C_{L\alpha} \alpha$ blended with a flat-plate post-stall model through a sigmoid function, which allows the simulation to remain well-defined if the angle of attack exceeds stall [5]. Drag follows a parabolic polar, $C_D = C_{D0} + k C_L^2$. The pitching moment coefficient is $C_m = C_{m0} + C_{m\alpha} \alpha + C_{mq} (q c / 2V) + C_{m\delta e} \delta_e$. The rotors, pusher, elevator and rotor-moment channel are modeled as first-order lags with time constants of 0.08, 0.15, 0.05 and 0.08 s, and with the saturation limits in Table 2.

2.4 Simulation environment and reproducibility

The model of Section 2.2 corresponds one-to-one to a Simulink diagram built from an Interpreted MATLAB Function (or MATLAB Function) block for the equations of motion, Integrator blocks for the states, Transfer Function blocks for the actuator lags, and PID Controller blocks with anti-windup for the loops of Section 3. The numerical results reported in this paper were generated with an equivalent numerical implementation (fixed-step fourth-order Runge-Kutta integration at 100 Hz for the time responses, 50 Hz inside the optimizer, with first-order filtered derivative action at $N = 20$ and Gaussian sensor noise), and the same equations and parameters are supplied here so that the model can be rebuilt in Simulink and the results reproduced. Appendix A lists the MATLAB commands used for the corresponding linearization, margin and optimization steps.

3 TRANSITION STRATEGY AND CONTROL ARCHITECTURE

3.1 Transition schedule

The transition is commanded as a smooth airspeed ramp $V_{\text{ref}}(t) = 22 \text{ m/s} \cdot s((t - 2)/14)$, where $s(x) = 3x^2 - 2x^3$ is a smooth-step, starting after a 2 s hover and ending at 22 m/s at $t = 16$ s, while the altitude reference is held constant at 50 m. Two smooth schedules, both functions of the measured airspeed, distribute the work between actuators:

- **Lift share** $w_v(V) = 1 - s((V - 5)/13)$: fraction of the weight carried by the lift rotors. Rotors carry the full weight below 5 m/s and are fully withdrawn at 18 m/s, which is 1.6 times the clean stall speed of about 11.4 m/s.
- **Elevator authority share** $w_e(V) = s((V - 6)/8)$: the commanded pitch moment is allocated as $\delta_e = w_e M_c / (\bar{q} S c C_{m\delta e})$ to the elevator and $(1 - w_e) M_c$ to the rotors as a differential moment M_r , using a floor on $\bar{q}$ to avoid division by a vanishing dynamic pressure.

3.2 Cascade of PID loops

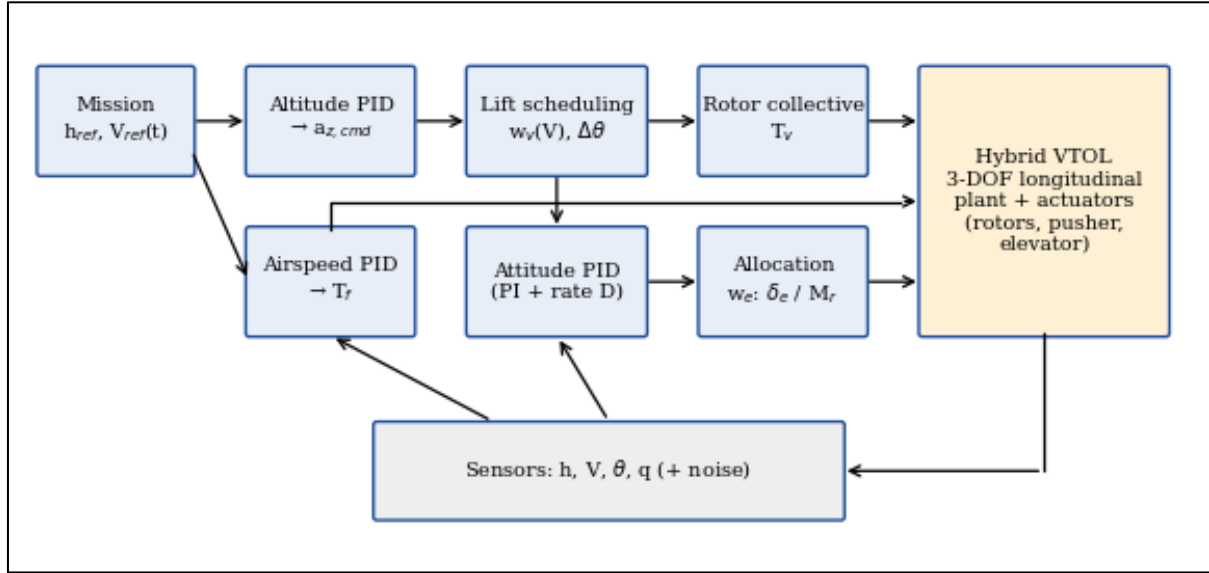


Figure 1: Control architecture of the hybrid VTOL UAV: three PID loops combined with airspeed-based scheduling and control allocation.

Three PID loops (Fig. 1) are used. The **airspeed loop** commands pusher thrust, $T_f = K_{p,V} e_v + K_{i,V} \int e_v dt - K_{d,V} \dot{V}_f$, clamped to $[0, 40]$ N, with conditional-integration anti-windup. The **altitude loop** produces a commanded vertical acceleration $a_z = K_{p,h} e_h + K_{i,h} \int e_h dt - K_{d,h} \dot{h}_f$ limited to ± 4 m/s². This is converted to a collective rotor thrust $T_v = w_v m (g + a_z) / \cos \theta$ and, in the wing-borne regime, to a pitch-reference increment $\Delta\theta = (1 - w_v) m a_z / (\bar{q} S C_{L\alpha})$, the inverse of the lift curve slope, so that the same loop gain is retained from hover to cruise. The **attitude loop** computes the virtual pitch moment

$$M_c = I_{yy} [K_{p,\theta} (\theta_{ref} - \theta) + K_{i,\theta} \int (\theta_{ref} - \theta) dt - K_{d,\theta} \dot{\theta}] \quad (8)$$

with $\theta_{ref} = \theta_{trim}(V) + \Delta\theta$, where $\theta_{trim}(V)$ is a trim-pitch look-up table obtained from the trim solution of Section 4. Derivative action acts on the measured pitch rate (rate gyro), which avoids derivative kick on reference changes. Scaling the moment by I_{yy} makes the attitude gains independent of the vehicle inertia: in hover, $K_{p,\theta}$ and $K_{d,\theta}$ set the natural frequency and damping of the closed pitch loop as $\omega_n = \sqrt{K_{p,\theta}}$ and $\zeta = K_{d,\theta} / (2\sqrt{K_{p,\theta}})$.

4 STABILITY ANALYSIS

4.1 Trim and linearization

For each airspeed V the level-flight equilibrium ($\alpha = \theta, h = 0$) is found by solving $\dot{u} = \dot{w} = \dot{q} = 0$ for α, T_f and δ_e , with the lift-rotor thrust fixed by the schedule $T_v = w_v m g$. The nonlinear model is then linearized numerically (central differences) in the state $x = [u, w, q, \theta]^T$ with the actuator outputs held at their trim values [4], [13]. Table 3 lists the trim conditions. At 5 m/s the trim pusher thrust is slightly negative (−2.4 N) because rotor drag and the nose-down trim attitude exceed the available forward force; the controller clamps T_f at zero in that regime, so this point should be read as a linearization point rather than an operating point.

Table 2: Trim conditions along the transition

V (m/s)	w_v	θ_{trim} (deg)	T_v (N)	T_f (N)
0	1.00	0.0	117.7	0.0
5	1.00	-2.8	117.7	-2.4
10	0.67	5.0	78.9	15.3
15	0.14	6.4	15.9	15.2
18	0.00	4.6	0.0	15.3
22	0.00	2.2	0.0	19.1

4.2 Open-loop modes

Table 3: Open-loop longitudinal eigenvalues (s^{-1})

V (m/s)	Eigenvalues $\sigma \pm j\omega$	Status
0	-1.64; $0.80 \pm j1.41$; -0.04	Unstable
5	$-1.39 \pm j2.28$; 0.99; -0.73	Unstable
10	$-2.49 \pm j4.13$; $0.01 \pm j0.39$	Unstable
15	$-3.65 \pm j6.10$; $-0.04 \pm j0.70$	Stable
18	$-4.35 \pm j7.32$; $-0.05 \pm j0.66$	Stable
22	$-5.30 \pm j8.94$; $-0.06 \pm j0.54$	Stable

The eigenvalues in Table 4 and Fig. 2(a) show the progression of the dynamics. At hover the plant has an unstable oscillatory pair at $+0.80 \pm j1.41 s^{-1}$ (period 4.5 s, time to double amplitude 0.87 s), produced by the coupling of pitch attitude, forward acceleration and the rotor flapping moment, together with two slow real modes. At 5 m/s a real unstable pole at $+0.99 s^{-1}$ is present while the wing is still too lightly loaded to help. From 10 m/s a well-damped short-period mode appears ($-2.49 \pm j4.13$ at 10 m/s, $\omega_n = 4.8$ rad/s, $\zeta = 0.52$), whose frequency grows with dynamic pressure to $-5.30 \pm j8.94$ at 22 m/s ($\omega_n = 10.4$ rad/s, $\zeta = 0.51$). The low-frequency oscillatory mode is slightly unstable at 10 m/s ($+0.012 \pm j0.39$), where rotors are still carrying 67% of the weight, and becomes lightly damped and stable above 15 m/s ($\zeta \approx 0.06$ to 0.12). Hence the transition requires active stabilization from hover to roughly 10 to 15 m/s, and the closed loop must also damp the lightly damped low-frequency mode throughout.

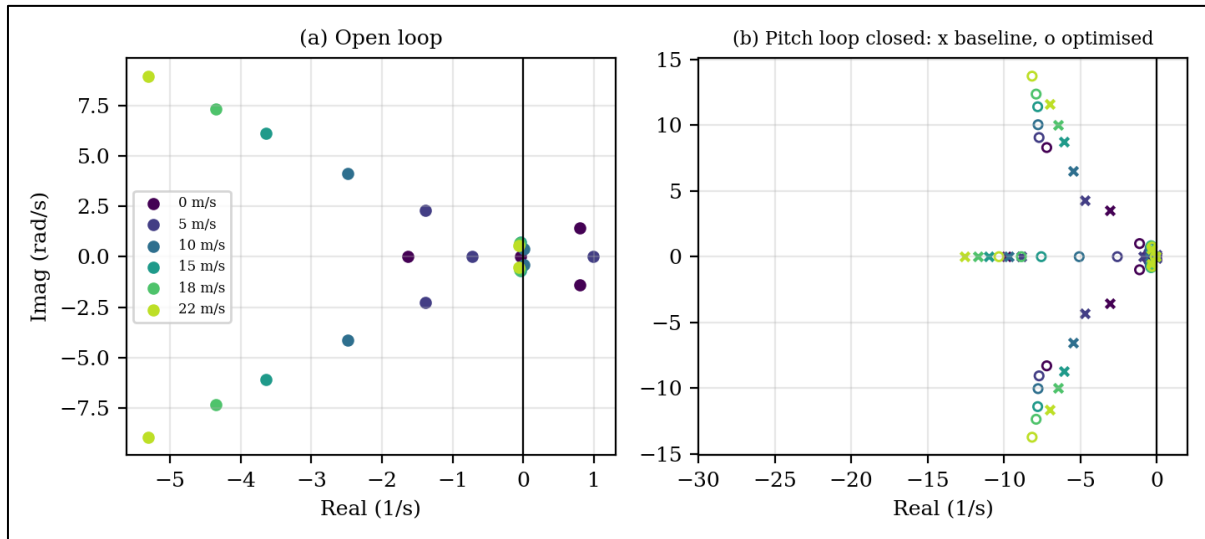


Figure 2: Longitudinal eigenvalues at six trim airspeeds: (a) open loop; (b) with the pitch loop closed for the baseline (x) and PSO-optimised (o) gains.

4.3 Pitch-loop stability margins

The pitch loop is analyzed by breaking it at the moment input. With the controller of Eq. (8) and the actuator and computation lag modeled as $1/(0.06 s + 1)$ with a 10 ms delay, the loop transfer function is $L(s) = I_{yy} (K_{p,\theta} + K_{i,\theta}/s + K_{d,\theta} s) G_{\theta M}(s) e^{(-0.01s)/(0.06s + 1)}$, where $G_{\theta M}(s)$ is the transfer function from pitch moment to pitch angle obtained from the linearized model [14]. Because the open-loop plant is unstable at low speed, the Nyquist criterion must be applied with the right-half-plane poles taken into account, and the phase margin at the highest gain crossover is reported together with the closed-loop eigenvalues. Results are given in Section 6.

5 PID GAIN OPTIMIZATION

5.1 Problem formulation

The nine gains $k = [K_{p,\theta}, K_{i,\theta}, K_{d,\theta}, K_{p,h}, K_{i,h}, K_{d,h}, K_{p,v}, K_{i,v}, K_{d,v}]$ are found by minimizing the cost

$$J(k) = \sum_c [0.3 \cdot ITAE_h + 8 \cdot ITAE_\theta + 25 \cdot \int |V_{ref} - V| dt + 200 \cdot |V(t_f) - V_{ref}| + 30 \cdot \max|e_h| + 0.5 \cdot J_u + P_\alpha] + 0.15K_{p,\theta} + 0.5K_{d,\theta} + 40 \cdot \max(0, 45^\circ - PM_{min}) \quad (9)$$

where the sum is over two scenarios (c): a calm run and a run with a 3 m/s vertical gust between $t = 8$ s and $t = 12$ s, both with sensor noise ($\sigma = 0.009$ rad/s on q , 0.0035 rad on θ , 0.1 m on h , 0.1 m/s on u and w). $ITAE_h$ and $ITAE_\theta$ are the integral of time-weighted absolute altitude and attitude-tracking errors, J_u is a control-activity measure (sum of squared elevator-command increments plus smaller weights on rotor moment and collective-thrust increments), P_α penalizes angle of attack above 12° while $V > 5$ m/s, the two gain terms discourage noise-amplifying designs, and PM_{min} is the minimum linear phase margin of the pitch loop evaluated at 0, 5 and 10 m/s, the airspeeds at which the plant is most difficult to stabilize. A run in which the pitch exceeds 69° or the altitude drifts by more than 60 m is assigned a cost of 10^6 .

5.2 Particle swarm optimization

PSO [11] was selected for its simplicity and its ability to handle a non-smooth, simulation-based cost with no gradient. The swarm has 20 particles and 30 iterations, with an inertia weight decreasing linearly from 0.9 to 0.4, cognitive and social coefficients of 1.6, and bound constraints on all gains (Table 5). The baseline gains were inserted as one particle in the initial swarm so that the optimizer could not return a result worse than the starting design. Each cost evaluation runs two 40 s simulations; the best cost and the gains are logged at every iteration (Fig. 3).

Table 4: PID gains: search bounds, baseline and pso-optimised values

Gain	Unit	Lower	Upper	Baseline	Optimised
$K_{p,\theta}$	s^{-2}	4.00	50.00	16.00	18.02
$K_{i,\theta}$	s^{-3}	0.00	12.00	4.00	12.00 *
$K_{d,\theta}$	s^{-1}	3.00	14.00	5.60	9.33
$K_{p,h}$	s^{-2}	0.20	3.00	1.00	1.41
$K_{i,h}$	s^{-3}	0.00	0.80	0.20	0.00 *
$K_{d,h}$	s^{-1}	0.50	5.00	1.80	0.50 *
$K_{p,v}$	N·s/m	2.00	10.00	6.00	10.00 *
$K_{i,v}$	N/m	0.50	5.00	3.00	5.00 *
$K_{d,v}$	N·s ² /m	0.00	3.00	0.00	0.00 *

* value at a search bound. The baseline attitude gains correspond to a hover natural frequency of 4 rad/s and damping ratio of 0.7; the baseline altitude and airspeed gains were tuned by hand.

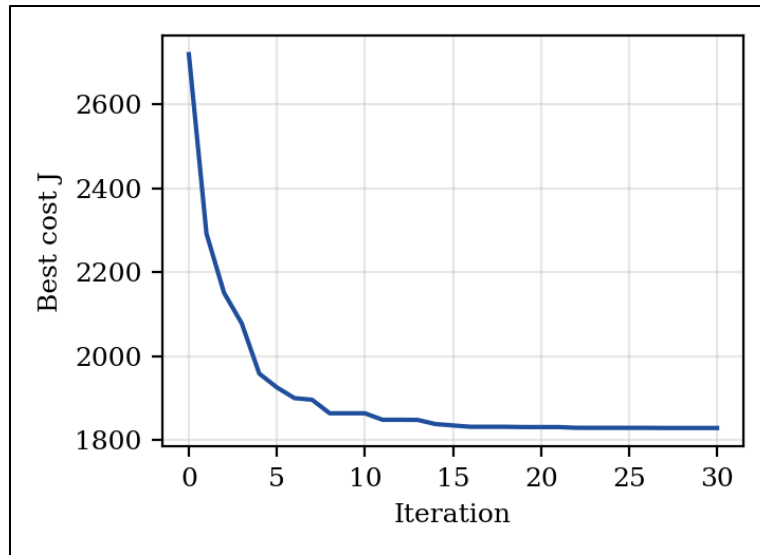


Figure 3: Convergence of the PSO best cost.

The optimizer drove $K_{i,\theta}$, $K_{p,v}$ and $K_{i,v}$ to their upper bounds and $K_{i,h}$, $K_{d,h}$ and $K_{d,v}$ to their lower bounds. In practice this means that the attitude-integral action and the airspeed loop would benefit from more authority than the bounds allow, which are in turn limited by actuator saturation and noise, whereas the altitude loop works best with proportional action and a small derivative term because the scheduled thrust feed-forward already supports the vehicle weight. Wider bounds should be examined only together with a model that includes actuator rate limits and realistic sensor dynamics.

5.3 A lesson on cost-function design

In earlier optimization runs without the phase-margin term, the gains reached very low altitude error but with a hover phase margin of only 1.9° . A second, equally instructive failure occurred when airspeed tracking was weakly weighted: the optimizer held altitude perfectly by never accelerating beyond 9 m/s, which is not a transition at all. These outcomes are typical of simulation-driven tuning, since a cost computed in the time domain with an ideal plant rewards bandwidth without regard to robustness. The final cost therefore includes a final-speed term and an explicit margin constraint.

6 RESULTS AND DISCUSSION

6.1 Stability margins

Table 5: Pitch-loop phase margin and crossover frequency

V (m/s)	PM baseline (deg)	PM optimised (deg)	ω_c baseline (rad/s)	ω_c optimised (rad/s)
0	39.5	45.0	5.77	8.30
2	47.1	49.1	5.85	8.41
4	52.8	52.6	6.00	8.51
6	59.6	56.2	6.22	8.73
8	67.8	60.0	6.46	9.06
10	77.9	64.8	6.63	9.29
12	92.1	70.2	6.54	9.53
14	116.9	77.1	1.78	9.65
16	111.1	84.7	1.53	9.77
18	106.2	96.6	1.33	9.53
20	102.0	97.4	1.18	1.25
22	98.5	90.0	1.05	1.12

Table 5 and Fig. 4 show that both controllers stabilize the pitch loop across the whole transition. The baseline has a phase margin of 39.5° at hover, below the 45° target used for the optimized design, and a margin that rises to 117° to 111° at 14 to 16 m/s and about 99° at 22 m/s, where aerodynamic damping dominates. The optimized design has a minimum phase margin of 45.0° at hover, i.e. it sits on the 45° constraint, and its margin increases monotonically with airspeed to 85° at 16 m/s and 90° at 22 m/s. The optimized design trades some margin at intermediate speeds (for example 60° against 68° at 8 m/s) for a higher low-speed crossover frequency (8.3 to 9.5 rad/s between hover and 18 m/s against 5.8 to 6.5 rad/s for the baseline up to 12 m/s), the combined effect of a larger rate-feedback gain ($K_{d,\theta} = 9.3$ against 5.6) and a higher integral gain; both designs have a crossover of about 1.1 rad/s at 22 m/s. The largest real part of the closed-loop pitch-loop eigenvalues across the six trim speeds is -0.032 s^{-1} for the optimized design and -0.020 s^{-1} for the baseline, so the pitch loop is stable at all analyses airspeeds; the slowest closed-loop modes are lightly damped low-frequency modes with real parts close to zero, which the pitch loop alone does not damp strongly and which are left to the altitude and airspeed loops.

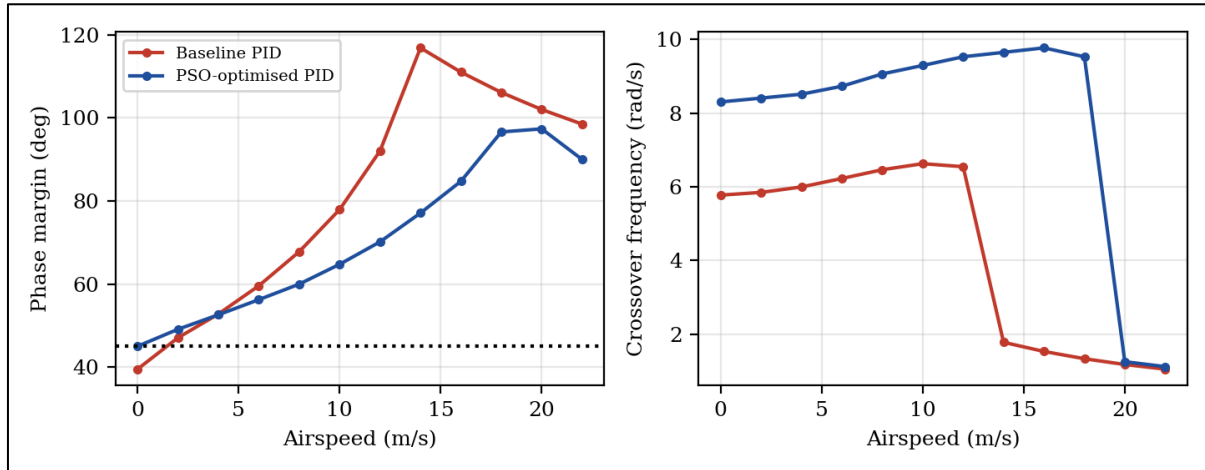


Figure 4: Pitch-loop phase margin and gain-crossover frequency versus airspeed. The dotted line marks the 45° design constraint.

6.2 Transition time responses

Table 6: Transition performance (nominal and 3 m/s gust cases)

Metric	Baseline nominal	Optimised nominal	Baseline gust	Optimised gust
Peak altitude deviation (m)	7.96	3.71	11.31	5.21
ITAE altitude ($\text{m}\cdot\text{s}^2$)	2310	366	4442	637
Altitude settling time, $\pm 0.5 \text{ m}$ (s)	40.0†	26.5	39.0†	28.0
Airspeed overshoot (%)	5.8	2.6	7.1	1.8
Time to 90% of V_{ref} after ramp start (s)	11.6	11.2	12.9	12.3
Peak pitch angle (deg)	6.3	5.9	12.0	8.7
Peak angle of attack, $V > 5 \text{ m/s}$ (deg)	5.4	5.4	5.3	6.3
Control activity J_u	241	38	257	39

† not settled within the 0.5 m band at the end of the 40 s simulation.

In the nominal transition (Fig. 5, Table 6), the largest altitude errors occur near the end of the speed ramp ($t \approx 14$ to 17 s , $V \approx 21$ to 23 m/s), when the rotors are off and altitude must be held through the pitch channel as the pusher completes the acceleration. The baseline sags to a peak deviation of 8.0 m. The optimized controller reduces this to 3.7 m, the altitude ITAE by a factor of 6.3, and the airspeed overshoot from 5.8% to 2.6%, with a control-activity index 6.3 times lower, which indicates that the improvement does not come at the expense of more aggressive actuator action. The angle of attack stays below 5.4° , far from the 16° stall angle, in both cases. The time to reach 90% of the reference airspeed is nearly the same for both controllers (about 11 to 12 s after the ramp starts) because it is set mainly by the pusher thrust limit and the ramp profile, not by the PID gains. The altitude settling time to $\pm 0.5 \text{ m}$ remains long (26 s) because the optimized altitude loop has no integral action ($K_{i,h}$ at its lower bound), so the final residual error decays

slowly through the rotor/wing hand-over; if tight settling is a requirement, a non-zero $K_{i,h}$ with an explicit settling-time term should be added to the cost.

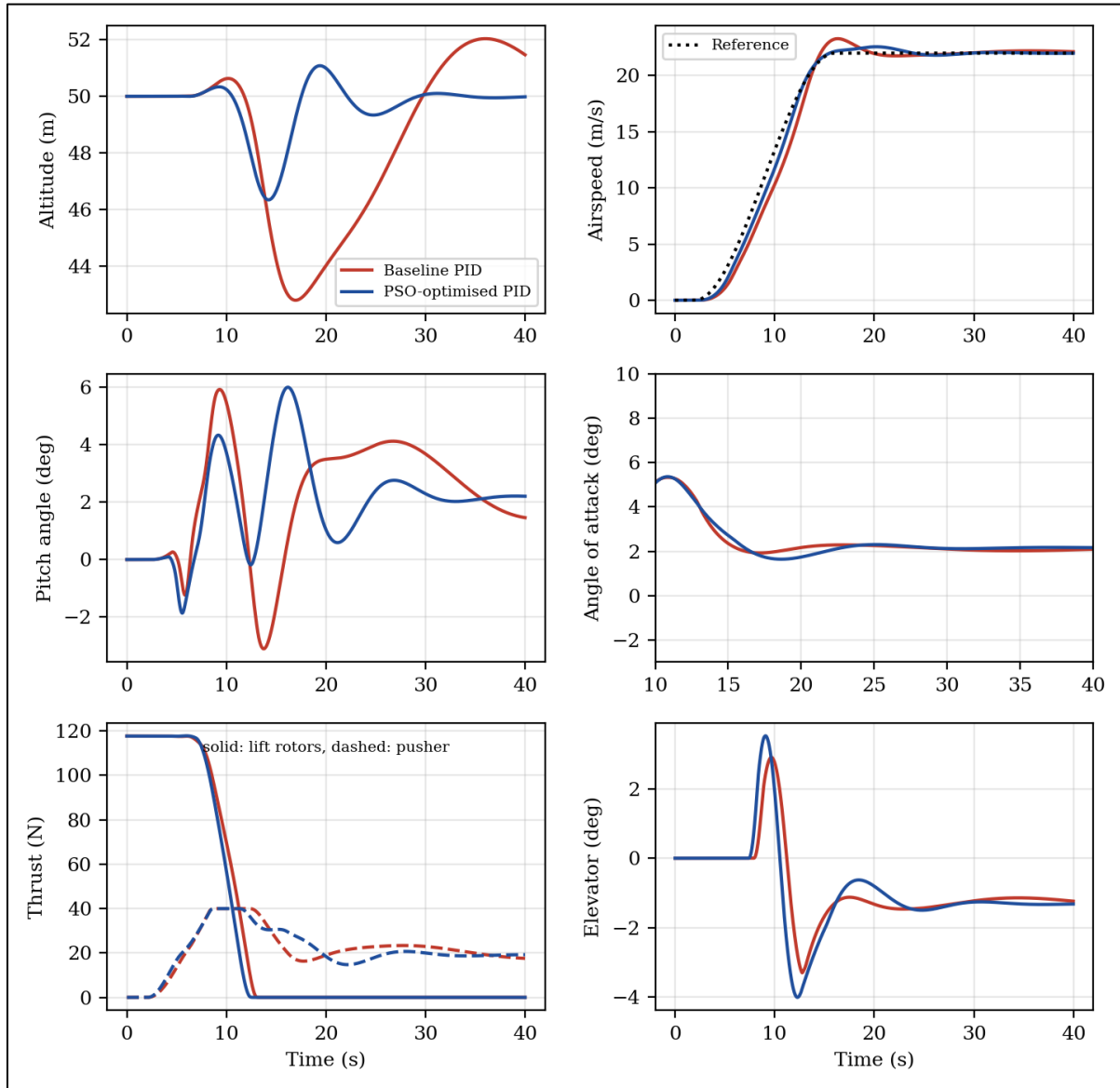


Figure 5: Hover-to-forward-flight transition: altitude, airspeed, pitch angle, angle of attack, thrust and elevator for the baseline and the PSO-optimised PID controllers (no gust, with sensor noise disabled for plotting).

6.3 Gust rejection

The gust is applied while the airspeed rises from about 6 to 15 m/s, when the rotors and the wing both carry significant load and the vehicle is most sensitive; the largest altitude deviation, however, occurs after the gust has ended, as the vehicle approaches cruise speed ($t \approx 15$ s for the optimized controller and $t \approx 20$ s for the baseline). As shown in Fig. 6 and Table 7, the optimized controller limits the peak altitude deviation to 5.2 m, against 11.3 m for the baseline, and the peak pitch angle to 8.7° against 12.0° . The optimized controller does permit a slightly higher peak angle of attack (6.3° against 5.3°), which is acceptable because it is still far from stall.

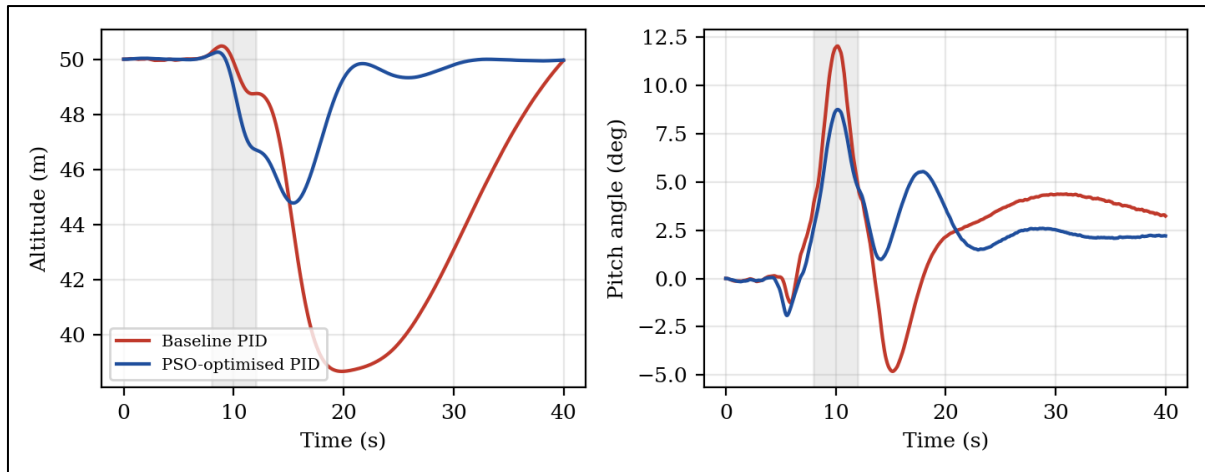


Figure 6: Altitude and pitch response to a 3 m/s vertical gust between $t = 8$ s and 12 s (shaded), applied while the vehicle is in transition.

6.4 Robustness to parameter uncertainty

To assess robustness to modeling error, 40 Monte Carlo runs were made with the plant perturbed uniformly in the ranges: mass 0.90 to 1.15 times nominal (the controller still assumes the nominal mass), $C_{m\alpha}$ between 0.6 and 1.0 times nominal (a reduced static margin), $C_{L\alpha} \pm 10\%$, $I_{yy} -10\%$ to $+20\%$, and $C_{D0} -10\%$ to $+30\%$, in each case together with the gust of Section 6.3 and a different sensor-noise seed.

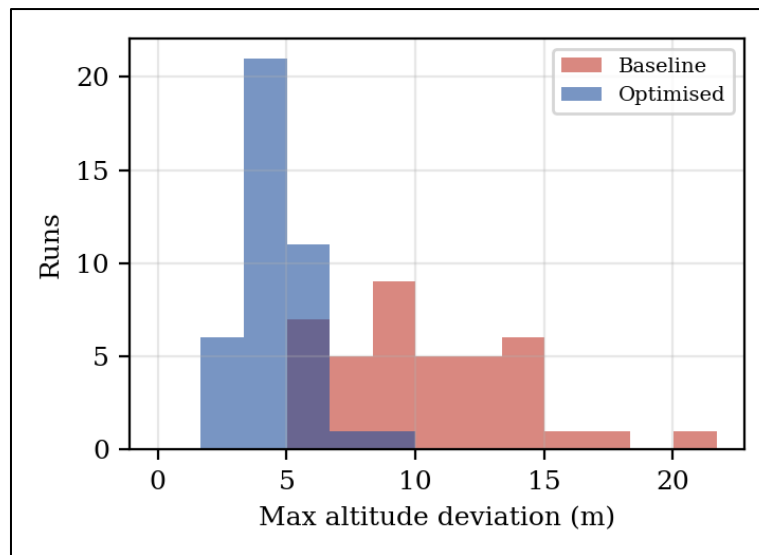


Figure 7: Distribution of the peak altitude deviation over 40 Monte Carlo runs.

Table 7: Monte carlo robustness results (40 runs)

Metric	Baseline	Optimised
Stable runs	40/40	40/40
Mean peak altitude deviation (m)	10.5	4.6
95th-percentile deviation (m)	15.5	6.5
Worst-case deviation (m)	20.7	8.6
Runs with deviation < 5 m	0/40	27/40

Both controllers remained stable in all runs (Table 7), so the improvement is in performance rather than basic stability. The optimized controller reduced the mean peak deviation from 10.5 m to 4.6 m and the worst case from 20.7 m to 8.6

m, with 27 of 40 runs inside 5 m against 0 of 40 for the baseline. The gains were optimized on the nominal model only, so this result indicates that the improvement is not highly sensitive to the modeled uncertainty, but it is not a guarantee: the uncertainty ranges are modest and uncertainty in the rotor model (k_{rd} , k_n), which dominates the low-speed regime, was not varied.

6.5 Limitations

- The model is longitudinal only. Lateral-directional coupling, rotor torque, yaw control and wind shear during transition are not represented.
- Rotor effects are represented by two lumped coefficients (momentum drag and flapping moment) with values chosen to be representative of a vehicle of this class, not identified from flight or wind-tunnel data; the low-speed stability results in particular depend on k_n .
- The optimization uses a single run per particle and one noise realization per scenario, a small swarm and 30 iterations; repeated runs with different seeds and a comparison with other optimizers, such as differential evolution [12], would give statistical confidence in the gains.
- The phase-margin constraint is applied to the inner pitch loop only; the outer altitude and airspeed loops were verified in simulation but not by a formal loop-shaping analysis.
- Results are from a numerical model; they have not been validated on hardware. Hardware-in-the-loop and flight tests are the necessary next steps.

NOMENCLATURE

Symbol	Description
m, I_{yy}	Mass (kg), pitch moment of inertia ($\text{kg}\cdot\text{m}^2$)
S, c	Wing reference area (m^2), mean aerodynamic chord (m)
u, w, q, θ	Body-axis velocities (m/s), pitch rate (rad/s), pitch angle (rad)
h, V, α	Altitude (m), airspeed (m/s), angle of attack (rad)
T_v, T_f	Total lift-rotor thrust and pusher thrust (N)
δ_e, M_r	Elevator deflection (rad), rotor differential pitch moment ($\text{N}\cdot\text{m}$)
w_v, w_e	Rotor lift share and elevator authority share (-)
C_L, C_D, C_m	Lift, drag and pitching-moment coefficients
K_p, K_i, K_d	Proportional, integral and derivative gains
PM, ω_c	Phase margin (deg), gain-crossover frequency (rad/s)
ITAE	Integral of time-weighted absolute error

7 CONCLUSIONS

A nonlinear longitudinal model of a quad-plane hybrid VTOL UAV was developed and linearized across the hover-to-forward-flight transition. The analysis shows an unstable oscillatory pair at hover and an unstable real pole near 5 m/s that are eliminated by increasing dynamic pressure, so the transition requires scheduled, active stabilization up to roughly 10 to 15 m/s. A cascade of three PID loops with airspeed-scheduled lift sharing and pitch-control allocation was tuned by particle swarm optimization with an explicit 45° phase-margin constraint. Relative to a hand-tuned baseline, the optimized controller reduced the peak altitude deviation in transition from 8.0 m to 3.7 m and in a 3 m/s gust from 11.3 m to 5.2 m, reduced airspeed overshoot from 5.8% to 2.6%, and maintained a pitch-loop phase margin of at least 45° at all airspeeds. The Monte Carlo study showed that these gains carry over to a perturbed vehicle. Two practical lessons follow: a margin constraint must be part of the cost to prevent a time-domain optimizer from returning fragile designs, and the cost must penalize failure to complete the transition, otherwise the optimizer finds trivial solutions. Future work includes extending the model to six degrees of freedom, adding actuator rate limits and rotor-model uncertainty, comparing optimizers, and hardware-in-the-loop validation.

STATEMENTS ON COMPLIANCE WITH ETHICAL STANDARDS

Acknowledgments

I sincerely appreciate everyone for your support, contributions, and commitment.

Disclosure of conflict of interest

There is no conflict of interest to be disclosed.

REFERENCES

- [1] A. S. Saeed, A. B. Younes, C. Cai, and G. Cai, "A survey of hybrid unmanned aerial vehicles," *Progress in Aerospace Sciences*, vol. 98, pp. 91–105, 2018.
- [2] G. J. J. Ducard and M. Allenspach, "Review of designs and flight control techniques of hybrid and convertible VTOL UAVs," *Aerospace Science and Technology*, vol. 118, p. 107035, 2021.
- [3] N. T. Hegde, V. I. George, C. G. Nayak, and A. C. Vaz, "Transition flight modeling of a fixed-wing VTOL UAV," *Journal of Intelligent & Robotic Systems*, vol. 100, 2020.
- [4] B. L. Stevens, F. L. Lewis, and E. N. Johnson, *Aircraft Control and Simulation*, 3rd ed. Hoboken, NJ, USA: Wiley, 2015.
- [5] R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*. Princeton, NJ, USA: Princeton Univ. Press, 2012.
- [6] B. Etkin and L. D. Reid, *Dynamics of Flight: Stability and Control*, 3rd ed. New York, NY, USA: Wiley, 1996.
- [7] R. C. Nelson, *Flight Stability and Automatic Control*, 2nd ed. New York, NY, USA: McGraw-Hill, 1998.
- [8] K. J. Åström and T. Häggglund, *PID Controllers: Theory, Design, and Tuning*, 2nd ed. Research Triangle Park, NC, USA: ISA, 1995.
- [9] K. H. Ang, G. Chong, and Y. Li, "PID control system analysis, design, and technology," *IEEE Trans. Control Systems Technology*, vol. 13, no. 4, pp. 559–576, Jul. 2005.
- [10] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Trans. ASME*, vol. 64, pp. 759–768, 1942.
- [11] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, Perth, Australia, 1995, pp. 1942–1948.
- [12] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [13] M. V. Cook, *Flight Dynamics Principles*, 3rd ed. Oxford, U.K.: Butterworth-Heinemann, 2013.
- [14] K. Ogata, *Modern Control Engineering*, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2010.

APPENDIX A. MATLAB/SIMULINK IMPLEMENTATION NOTES

The following MATLAB commands outline the corresponding analysis steps for a Simulink model named `hybrid_vtol_long` whose states are `[u w q θ]` and whose input is an additive pitch moment. They are a template for rebuilding the study and have not been run as part of the work reported here; port names and indices should be adapted to the model.

```
% 1. Trim and linearise at airspeed V (e.g. with findop / operspec)
```

```
[A,B,C,D] = linmod('hybrid_vtol_long', x_trim, u_trim);
```

```
eig(A) % open-loop modes (Table 4)
```

```
% 2. Pitch-loop margins
```

```
Gth = tf(ss(A,B,C(4,:),0)); % moment -> pitch angle
```

```

Cpid = Iyy*tf([Kd Kp Ki],[1 0]); % Iyy*(Kp + Ki/s + Kd*s)
Act = tf(1,[0.06 1]); % actuator and computation lag
L = Cpid*Act*Gth; L.InputDelay = 0.01;
[Gm,Pm,Wcg,Wcp] = margin(L); % check Pm >= 45 deg

% 3. PSO tuning of the nine gains
lb = [4 0 3 0.2 0 0.5 2 0.5 0];
ub = [50 12 14 3 0.8 5 10 5 3];
opts = optimoptions('particleswarm','SwarmSize',20,'MaxIterations',30);
k = particleswarm(@vtol_cost, 9, lb, ub, opts); % vtol_cost runs sim() and returns J

```

The cost function vtol_cost should run the two scenarios (calm and gust), assemble J as in Eq. (9), and return a large penalty if the simulation diverges.